Object Clustering with Dirichlet Process Mixture Model for Data Association in Monocular SLAM

Songlin Wei, Guodong Chen*, Wenzheng Chi*, Zhenhua Wang, and Lining Sun

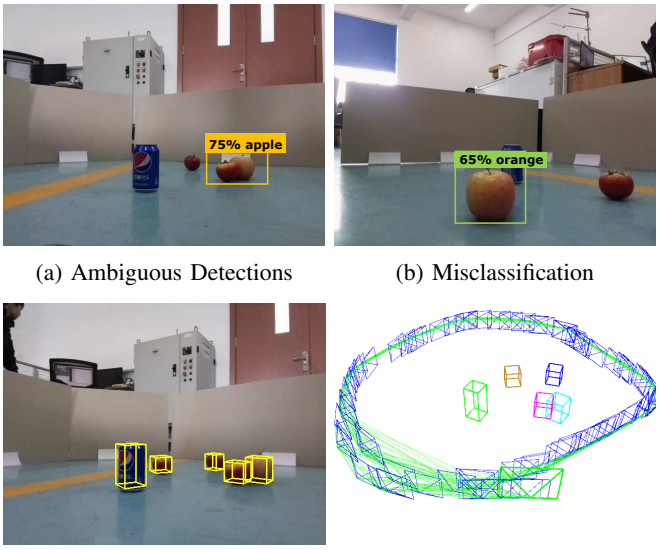
***Abstract*—Semantic SLAM with a monocular camera is particularly attractive because of the deployment simplicity and economic availability. Data association problem which assigns unique identities for objects shown in multiple frames plays a fundamental role in semantic slam. Previous prevalent methods which mainly focused on associating geometric KeyPoints are no longer suitable. Some naive methods that rely on object distance or 2D/3D Intersection over Union are also vulnerable when occlusions happen. In this paper, we propose a novel data association method for cuboid landmarks based on Dirichlet Process Mixture Model. By jointly considering object class, position, and size, our method can perform data association robustly. We evaluated our method in simulated datasets, public benchmark KITTI and on a real robot in an office environment. Experimental results show that our method not only associates cuboids robustly but also achieves SOTA pose estimation accuracy in monocular SLAMs.**

***Index Terms*—Cuboid object detection, Data association, Monocular SLAM, Semantic SLAM**

I. Introduction

SIMULTANEOUS localization and mapping (SLAM) is a fundamental problem being researched for mobile robots [1] [2], autonomous driving vehicles [3] [4], and industrial manipulators [5] [6]. In recent years, computer vision community has developed a mount of object detectors using monocular images. With the development of deep learning technology, monocular semantic slam that combines SLAM with object detection, recognition and reconstruction has become the most active research field. Instead of representing the environment with only geometrical shapes, semantic slam can also attach semantic information to the map, for example, object categories, functional usages, semantic relationships, etc. Object categories can be obtained through object detectors like [7], then object functionalities and relationships can further be acquired through a knowledge base [8]. Theoretically, this neglected semantic information might make object data association easier [1].

Correct data association which matches detected objects to map landmarks is critical. Ambiguous data association could lead to significant localization drift and even false loop detection, which would lead to catastrophic mapping failure [2]. Due to the lack of accuracy when estimating depth with a single camera, 3D object detection using a monocular image is challenging [3]. In monocular SLAM, objects are often represented as cuboids. The difficulties of associating cuboids are twofold. First, the cuboid detection with monocular image is an ill-posed problem and thus could easily produce suboptimal results. Second, in scenes where massive occlusions happen, the naive method using 2D Intersection over Union (IoU) or distance only could make false associations. Consequently, the erroneous constraints provided by these false associations could be detrimental to the tracking.



(a) Ambiguous Detections (b) Misclassification

(c) Correct Cuboid Detections (d) Succesful Mapping Results

Fig. 1: (a) Two apples are closely placed together, the 2D object detector failed to separate them. (b) The object detector occasionally produces a false class prediction. For example, it outputs an orange label for an apple. (c) Our data association method can handle the aforementioned situations and yields the right cuboid detections. (d) The optimized camera trajectory along with detected cuboid classes, shapes, and positions.

To achieve robust data association in monocular SLAM, we propose a novel method based on the Dirichlet process clustering approach and further unifies object location, shape, and semantic label information in the data association. Our main contributions are listed as follows:

- Based on the clustering with Dirichlet process mixture model, we apply the probabilistic cuboid measurement to associate objects. The algorithm can associate objects robustly in ambiguous environments.

* Corresponding author

Research was partially supported by the National Key R&D Program of China grant #2019YFB1310201 and was partially supported by National Science Foundation of China grant #61903267.

Songlin Wei, Guodong Chen, Wenzheng Chi, Zhenhua Wang, and Lining Sun are with the Robotics and Microsystems Center, School of Mechanical and Electric Engineering, Soochow University, Suzhou 215021, China {slwei, chenguodong, wzchi, wangzhenhua, lnsun}@suda.edu.cn

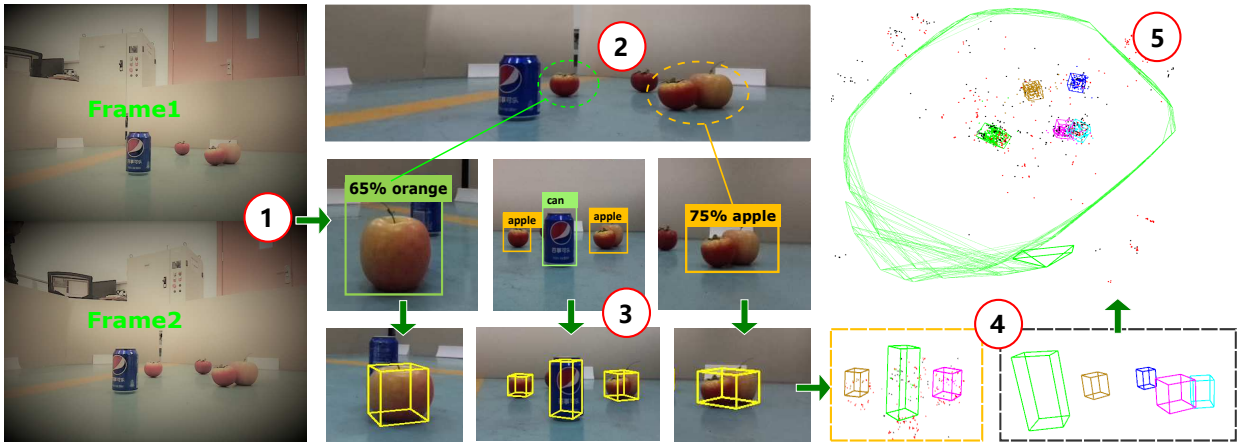


Fig. 2: Key steps of our system: (1) Visual odometry with consecutive frames. (2) Detect 2D bounding boxes with start-of-the-art detectors YOLO2. (3) Detect 3D cuboids using 2D boxes and odometry based in Vanishing Point technique. (4) Perform scale alignment with 3D map points and data association of cuboid landmarks. (5) Consistent mapping with semantic objects as 3D cuboids along with sparse map points.

- The cuboids are included as a new type of vertex in the pose graph of SLAM optimization. The cuboids not only can help reduce scale drift but also boost the performance of loop closing.

The rest of this paper is organized as follows. Section II reviews the related work and overviews how our approach contributes further. The proposed data association method is described in detail in Section III. We conduct a series of experiments and discuss the results in Section IV and finally draw conclusions in Section V.

II. Related Work

A. Object SLAM

Semantic information is demanded in SLAM systems to enable high-level user interaction and agent intelligence. Semantic segmentation and object detection are two common ways to produce semantics in the map. For example, Chen *et al.* [9] used a semantic segmentation network to label the projected image of the point cloud to create semantic maps. In this paper, we mainly focus on object detection. Recent researchers studied the integration of object detectors and SLAM. Pillai and Leonard [10] proposed to incorporate multi-view information of SLAM to help object recognition. Zhong *et al.* [11] tightly coupled object detector and SLAM to build an instance-level semantic map. By leveraging the object map, the object detector is improved under more challenging conditions. Xiao *et al.* [12] used object detector to eliminate the negative impact of dynamic objects, and the location accuracy is improved. Sharma *et al.* [13] proposed to reconstruct the environment with a graph of objects. A novel compositional rendering method is used to enable reliable frame-to-model RGB-D tracking. Nonetheless, the inclusion of object states into SLAM state estimation is not studied in these works.

Salas-Moreno *et al.* [5] and McCormac *et al.* [6] proposed RGB-D systems that detect objects by matching prior 3D models of known objects. However, the cost of build prior models is non-negligible. Naturally, academia has developed a considerable interest in monocular 3D object detection without prior models. Nicholson *et al.* [14] proposed to use dual quadrics to represent objects as ellipsoids, whereas our method use cuboids representation which is prevalent in autonomous driving like [3] [4] [15]. Yang and Scherer [15] proposed to generate cuboid landmarks with a 2D object detector and Vanishing Point(VP) technique. The generated cuboids are optimized together with camera poses and points. The authors proposed a simple method to associate cuboids by counting the shared map points. However, the method is not applicable in the situation where few map points are available in the map.

B. Data Association

To use objects in SLAM, the correspondences between measurements and objects must be established first. The problem is also known as data association, which is historically studied in the target tracking literature [16] [17] [18] [19]. Recently, Muresan and Nedevschi [17] proposed to use the Markov decision process to associate cuboids detected with LiDAR. Ikram and Ali [16] proposed a variant of the nearest neighbor data association method. However, unlike our work, the object data association results are not exploited in the SLAM context. The correspondence variables for data association are discrete whereas the camera poses and landmark positions are continuous. Bowman *et al.* [20] proposed to solve the two problems iteratively. They considered probabilistic data association (PDA) and leverage the Expectation-Maximization (EM) algorithm to find the correspondences between observed landmarks. Then, the poses of landmarks and camera trajectory were optimized with the correspondence variables fixed. We followed the two-phase iteration strategy but with a different cuboid data association approach. Elfring *et al.* [21] formally formulated data association problems in semantic SLAM, and applied Multiple Hypothesis Tracking (MHT) approaches. Wong *et al.* [22] had a detailed discussion of the limitations of MHT, and proposed a clustering-based approach to the problem. The clustering model with an unlimited number

of clusters is based on the Dirichlet process mixture model (DPMM). However, the approach was only applied to landmark estimations. Zhang *et al.* [23] also studied the data association problem with a hierarchical topic model based on the hierarchical Dirichlet Process. We novelly incorporate the cuboid measurement into the DPMM but without hierarchical structure. After the cuboid measurements are associated, we further jointly optimize cuboid positions and sizes, map points and camera poses.

III. METHODS

A. 3D Cuboid Estimation in monocular images

As aforementioned, we aim to detect objects without prior models in monocular images. We adopt cuboid representation for 3D objects in SLAM. The detected 3D cuboid is denoted as $C = \{R, L, S\}$, where $L = [x\, y\, z]^\intercal$ represents the center position of the cuboid and the object scale $S = [a\, b\, c]^\intercal$ is the length along each axis of the object coordinate frame. The object coordinate frame is built at the center of the cuboid. The object class is obtained through 2D object detector [7] while the object position, orientation and scale are obtained through a similar approach used in [15]. Other monocular 3D object detection methods like [4] can also be used.

B. Cuboid Data Association

1) Dirichlet Process and DPMeans method: In the semantic world modeling, most of the objects do not change in a short time, that is, most objects can be assumed to be static. Therefore, the time sequences of the measurements are irrelevant. The measurement-to-object association problem can be reduced to a clustering problem [24].

Now let $\mathcal{O} \triangleq \{(c^k, L^k, S^k)\}_{k=1}^{M}$ denote the set of all observations in a cluster, where c denotes object class and the observations in the cluster are indexed through $k = 1$ to M. The probability of a new object measurement (c, L, S) belongs to a certain cluster is:

$$\begin{aligned} &p(c, X, S|\,\mathcal{O}) \\ &= p(c\,|\,\mathcal{O})p(L\,|\,\mathcal{O})p(S\,|\,\mathcal{O}) \\ &= p(c\,|\,\{c^k\})p(L\,|\,\{L^k\})p(S\,|\,\{S^k\}\}). \end{aligned} \tag{1}$$

Notice that the cluster index is dropped for brevity, and the first equation assumes independence of object class, pose and shape, and the second equation follows from conditional independence properties. The first term on the right-hand side of (1) is the class predictive probability given all past observations of the same object. The second and third terms are predictive probabilities for pose and scale. Denote the class set as $\mathcal{C} = \{1, C\}$, the posterior probability of a cluster class to be c' is:

$$p(c'|\{c^k\}) \propto p(\{c^k\}|c')p(c') = \left[\prod_k p(c^k|c')\right] p(c'), \tag{2}$$

where $p(c')$ is the prior, and $p(c^k|c')$ is the class measurement probability. Then the class predictive probability can be calculated as:

$$p(c|\{c^k\}) = \sum_{i=1}^{C} p(c|c_i')p(c_i'|\{c^k\}). \tag{3}$$

Again, the $p(c|c_i')$ is the class measurement probability. $p(c_i'|\{c^k\})$ is the class posterior probability from (2). Under the static world assumption, the measurements of object location $[x\, y\, z]^\intercal$ and scale $[a\, b\, c]^\intercal$ both follow Gaussian distributions with unknown means and covariance. To simplify the instruction of conjugate prior of the normal distribution, we further assume that the x, y, z components of the cuboid center and a, b, c components of the cuboid scale are independent:

$$p(L|\{L^k\}) = p(x|\{x\}^k)p(y|\{y\}^k)p(z|\{z\}^k) \tag{4}$$

$$p(S|\{S^k\}) = p(a|\{a\}^k)p(b|\{b\}^k)p(c|\{c\}^k) \tag{5}$$

Each observation is assumed to be a single variable Gaussian distribution as illustrated in Fig. 3. Rather than simply assuming a fixed covariance of the measurements, we use a standard conjugate prior for Gaussian noise model with unknown mean and precision, namely the $NormalGamma(l, \tau; \lambda, \nu, \alpha, \beta)$ distribution. It is a continuous probability distribution with two variables and four parameters. The marginal distribution of the mean l is a non-standard t-student distribution:

$$p(l|\{x\}; \lambda, \nu, \alpha, \beta) = t\left(l; 2\alpha', \nu', \sqrt{\frac{\beta'}{\lambda'\alpha'}}\right). \tag{6}$$

According to [25], the posterior predictive distribution for the next location observation x can be acquired as:

$$p(x|\{x\}; \lambda, \nu, \alpha, \beta) = \frac{1}{\sqrt{2\pi}} \frac{\beta^{-\alpha^-}}{\beta^{+\alpha^+}} \frac{\lambda^-}{\lambda^+} \frac{\Gamma(\alpha^+)}{\Gamma(\alpha^-)}, \tag{7}$$

where the hyperparameters with $-$ superscripts are previous values exclude x, and $+$ are updated values with the current n observations of x according to the following updating rule in (8)-(11):

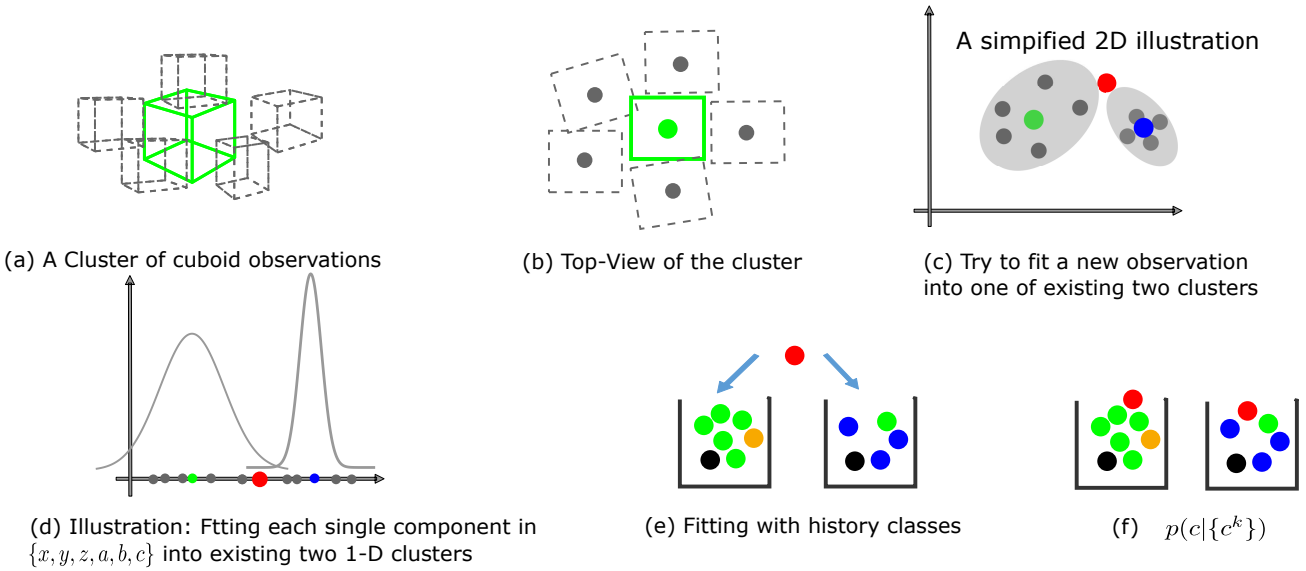
$$\lambda' = \lambda + n, \tag{8}$$

$$\nu' = \frac{\lambda}{\lambda + n}\nu + \frac{n}{\lambda + n}\hat{\mu}, \tag{9}$$

$$\alpha' = \alpha + \frac{n}{2}, \tag{10}$$

$$\beta' = \beta + \frac{1}{2}\left(n\hat{s^2} + \frac{\lambda n}{\lambda + n}(\hat{\mu} - \nu)^2\right), \tag{11}$$

where $\hat{\mu}$ and $\hat{s^2}$ are the sample mean and covariance, respectively. The probability of a new cuboid measurement k at time t fitting into an existing cluster can be calculated efficiently by substituting each component with (7) into (4)(5), and further substitute (3)(4)(5) into (1). Then, the measurement is assigned to the cluster j which has maximum probability, or a new cluster is created if the maximum probability is below some threshold $\mathbf{\Upsilon}$. The association result denoted as a pair $(z_{t,k}, j)$ is then added to the data association set $\mathbf{D}$.



(a) A Cluster of cuboid observations

(b) Top-View of the cluster

(c) Try to fit a new observation into one of existing two clusters

(d) Illustration: Ftting each single component in $\{x, y, z, a, b, c\}$ into existing two 1-D clusters

(e) Fitting with history classes

(f) $p(c|\{c^k\})$

Fig. 3: Here is a brief illustration of how the posterior predictive probability of fitting a new cuboid observation into existing clusters is calculated: (a) illustrates several cuboid observations forming a cluster due to perception noises. (b) Top-View of the cluster with dots represents the cuboid center positions. (c) A simplified 2D illustration of how to fit a new cuboid (represented as a red dot) into each of the exiting two clusters modeled as 2D Gaussian distributions centered at green and blues dots respectively. (d) We further assume each component of location (x,y,z) and shape scales (a,b,c) are independent, then each component is fitting into 1D Gaussian distribution with unknown mean and variance. Finally, (e) and (f) show how the class of the new cuboid is fitting into history observed classes by the analogy of putting a new colored ball into two each urn with noises (represented as black and yellow dots).

2) Robust SLAM: We maintain an object belief propagation process to evaluate the belief of each cluster of measurements corresponding to an actual object. The intuition is that the more likely a cluster is an actual object, the more consistent the measurements are. We examined the consistency of the measurements with the object class, location, and scale of the cuboids. For each object, the final belief of an object given a cluster of observations can be acquired by:

$$bel(\mathcal{O}) = (1 - p(0|\{c^k\}) \prod_{\substack{x,y,z \\ a,b,c}} t(l; 2\alpha', \nu', \sqrt{\frac{\beta'}{\lambda'\alpha'}}) \tag{12}$$

The first term of the right-hand side is the true detection probability. The second term is from (6). And the hyperparameters of T-student distributions are different for each component of $\{x, y, z, a, b, c\}$. To perform Robust SLAM, only object whose belief is over a certain threshold $\mathbf{\Phi}$ is considered valid, and all observations of valid objects are collected as known data associations.

C. Pose Graph Optimization

With known data associations $\mathbf{D} = \{(z_{t,k}, j)\}$, all the variables to be optimized are now continuous. Given cuboid object observations $\mathbf{Z} = \{\{(Z_{t,k}, S_{t,k})\}_{k=1}^{K}\}_{t=1}^{T}$, where $Z_{t,k}$ represents the k-th measurement of the center of the cuboid at time t, we are now able to perform joint optimization of cuboid landmarks $\mathbf{L} = \{(L, S)_{j=1}^{M}\}$ and camera trajectory $\mathbf{X} = \{X_t\}_{t=1}^{T}$. The MAP estimation problem can be formulated as:

$$\mathbf{X}, \mathbf{L} = \underset{\mathbf{X},\mathbf{L}}{\operatorname{argmax}}\, p(\mathbf{X}, \mathbf{L}|\mathbf{D}, \mathbf{Z}) \tag{13}$$

When applying the Gaussian noise model to cuboid observations, the MAP estimation problem of (13) is equivalent to Bundle Adjustment(BA) and can be solved with the g2o framework [26]. The BA problem can be formulated as optimization of a non-linear least square summations:

$$\hat{\mathbf{X}}, \hat{\mathbf{L}} = \underset{\mathbf{X},\mathbf{L}}{\operatorname{argmin}} \sum_{t,k} \|e(Z_{t,k}; X_t, L_j)\|^2 + \sum_{i,j} \|e(S_j; P_i, L_j)\|^2 + \sum_{t,i} \|e(p_{t,i}; P_i, X_t)\|^2 \tag{14}$$

The $e(Z_{t,k}; X_t, L_j)$ is the camera-cuboid measurement error. $e(S_j; P_i, L_j)$ is the cuboid-point measurement error. It transforms the map point to the cuboid frame and punishes the cost when the map point is outside the cuboid. $e(p_{t,i}; P_i, X_t)$ is the canonical 3D map point to 2D keypoint re-projection error.

D. Final Algorithm

The whole algorithm is described in Algorithm 1. During the initialization step, all the locations of landmark observations $\mathbf{L}$ are dead-reckoned with odometry data $\mathbf{O}_{1:T}$. Then data association is carried out as described in section III-B1. Instead of a single-step process, we iterate the whole process alternating between data association and pose optimization, which can revise incorrect associations due to the accumulated

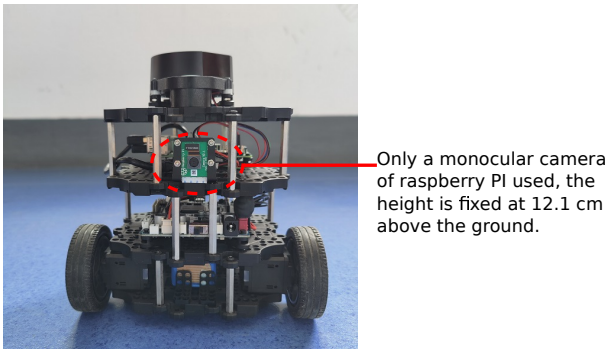


Fig. 4: Turtlebot3 Burger: a raspberry PI camera is mounted on the second Waffle-plate layer.

odometry error. For each new cuboid observation, the predictive probability $pred(i)$ conditioning on existing clusters $\mathcal{O}_i$ is calculated according to (1). If the maximum probability is below a certain pre-set threshold, a new cluster is created. Otherwise, this observation is assigned to the existing cluster $\mathcal{O}$. Repeat until all observations are processed. Next follows the pose optimization process. Object appearances including sizes and positions are updated after optimization. Specifically, the object position L is updated with the joint optimization result and the object size S is updated with (9) which is the mean of all observations. As mentioned in section III-B2, ambiguous data associations are discarded during the pose optimization process. Since the camera trajectory and landmark poses are converging at each iteration, more and more associations are found. The final number of objects (clusters) converges after a few iterations.

Algorithm 1: Dirichlet Process Object Clustering For Cuboid Data Association

Input: Odometry measurements $\mathbf{O}_{1:T}$, Cuboid measurements $\mathbf{Z}_{t,k}$
Output: Poses $\mathbf{X}_{1:T}$, Landmarks $\mathbf{L}_{1:M}$, Data Associations $\mathbf{D}$

Set $\mathbf{X}_0$ to Identity matrix representing world frame, Initialize $\mathbf{X}_{1:T}$, $\mathbf{L}$ with open loop predictions
while $\mathbf{X}, \mathbf{L}$ *not converged* **do**
  //phase one, data association:
  set $M = 0, \mathbf{L}$
  for *each* t *in* $1:T$, *each* k *in* $1:K_t$ **do**
    Compute the predictive probability of fitting current measurement into each cluster according to (1):
    for i *in* $1:M$ **do**
      $pred(i) = -log\, p(c_{t,k}, X_t, S_{t,k} | \mathcal{O}_i)$
    Assign $\mathbf{D}_{t,k}$ to be cluster m with maximum probability or create a new cluster if the best predictive probability is below some threshold:
    if $min(pred) > \Upsilon$ **then**
      $M = M + 1$
      create new cluster L_M with current measurement, and initialize $\beta_M = \beta_0$
    else
      $\mathbf{D}_{t,k} = \text{argmin}_i\, pred(i)$
      update existing cluster L_i:
      $\mathcal{O}_i = \mathcal{O}_i \cup Z_{t,k}$
      $\beta_i(c_{t,k}) = \beta_i(c_{t,k}) + 1$
      update hyperparameters $\lambda, \nu, \alpha, \beta$ according to (8)-(11)
      update $bel(i)$ according to (12)
  //phase two, joint optimization of landmark positions and trajectory
  for *each* l *in 1:M* **do**
    if $bel(l) < \Phi$ **then**
      mark L_l as invalid, and remove from $\mathbf{L}$ afterwards
  optimize $\mathbf{X}_{0:T}, \mathbf{L}$ with pose graph solver according to (14)
  update X, prepare for next loop

IV. Experiments

A. Simulated Dataset

We first demonstrate the algorithm with a simulated dataset. To compare our algorithm with [27], we followed a similar experiment method. In this simulation, 15 objects of 5 classes are randomly generated in a 2D plane. An 800 time-steps trajectory was simulated in a 10m$\times$10m room. During each time step, landmarks within 4m distance to the robot were observed. Different from the original paper, to demonstrate the effectiveness of the object belief propagation and robust data association, not only the Gaussian noises are added to odometry and landmark location measurements, but also the class prediction errors are simulated according to a Confusion Matrix shown in Table I.

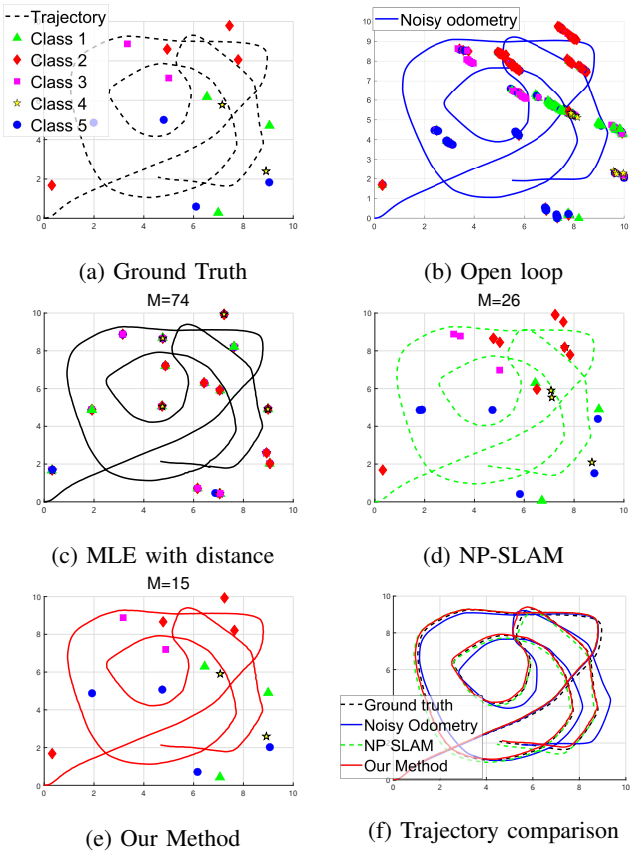
Fig. 5a shows the ground truth trajectory, and different marks and colors represent different classes of landmarks. Fig. 5b shows the robot's open-loop trajectory via dead reckoning. All the observed landmarks are also plotted without association. Due to the accumulated odometry noise, there is significant drift in robot location and so are the landmark positions. Besides, in some clusters, there are different classes of landmarks due to the designed class prediction Confusion Matrix in Table I. We compare our method with Maximum Likelihood Estimation(MLE) with distance only and Non-parametric SLAM (NP-SLAM) [27]. NP-SLAM considered both object class and position when performing data associations. The resulting trajectory is acceptable for MLE as shown in Fig. 5c, However, the estimated landmark number 74 is significantly larger than the ground truth 15. The reason is that ML lacks the mechanism to identify an object, it just associates each measurement to the nearest object. The NP-SLAM produces a reasonably accurate trajectory, but it also fails at estimating the right number of landmarks. Our method associates the right number of landmarks as shown in Fig. 5e. The comparison of the resulting trajectory is given in Fig. 5f.

B. Turtlebot3 Burger Collected Dataset

To execute our algorithm in a real-world environment, the robot Turtlebot3 Burger was adopted to collect a sequence of images in the office, as shown in Fig. 4. A raspberry PI camera

TABLE I: Confusion Matrix used in simulation

Actual Label	Predicted Label				
	class 1	class 2	class 3	class 4	class 5
class 1	0.8	0.06	0.04	0.04	0.01
class 2	0.06	0.78	0.05	0.04	0.02
class 3	0.09	0.03	0.77	0.05	0.01
class 4	0.09	0.03	0.06	0.75	0.02
class 5	0.04	0.03	0.07	0.02	0.79



(a) Ground Truth (b) Open loop

(c) MLE with distance (d) NP-SLAM

(e) Our Method (f) Trajectory comparison

Fig. 5: (a) shows the ground truth robot trajectory and classes and positions of landmarks. (b) shows the simulated noisy trajectory, and the open-loop perception of landmarks without data association. (c) shows the result of Maximum Likelihood. (d) shows the result of NP-SLAM [27]. (e) shows the data association result of our method. (f) compares trajectories of different methods.

was mounted at the second layer of the Waffle-Plate. The robot was driven manually to circle around clustered items placing on the ground, including tomatoes, pears, oranges, apples, and cans. 3 configurations are shown in first column of Fig. 8. The goal is to estimate the robot trajectory along with the classes, locations, and sizes of the items.

1) Implementation: Our system is built based on ORB-SLAM2 [28], the tracking module is used to provide initial odometry data. The data association part of Algorithm 1 is written in C++ and the local bundle adjustment is implemented with g2o framework [26]. After map initialization, the cuboids $\mathcal{C} = \{(c, L, S)\}$ detected in the first two frames are used to scale the map to align the cuboid detection scale with the monocular map scale. The scaled map initialization is shown in Fig. 2. It can be seen from the right part of the picture that

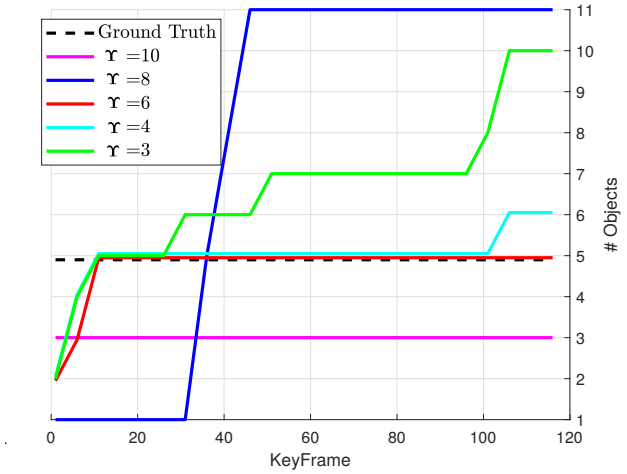


Fig. 6: Data association accuracy analysis with different $\boldsymbol{\Upsilon}$ settings in the Turtlebot3 indoor experiment. The ground truth number of objects is 5 shown in dotted black line. Best association accuracy is achieved when $\boldsymbol{\Upsilon}$=6.

most map points fit in the cuboid, which shows the success of scale alignment between cuboid detection and ORB-SLAM2 map creation.

2) Qualitative results: Fig. 1a shows an ambiguous bounding box produced by YOLO2 [7], the box contains two items but was predicted as one. In such cases, it is not surprising that the cuboid detection accuracy is affected, and even worse, a naive strategy could make a false association to this non-existing apple. As shown in Fig. 2, our resulting map is not affected by this ambiguous detection. This could be partially attributed to the effect of robust SLAM described in section III-B2 and partially attributed to the consideration of object sizes when matching to an existing cluster. As aforementioned, another challenge is class prediction error as shown in Fig. 1b, where the apple is marked as an orange. The results in Fig. 1d shows that our method is robust to low prediction error for the reason that we maintained an object belief when making association decisions.

3) Quantitative results for association: To quantitatively analyze the effectiveness of our data association algorithm, 3 different configurations of 11 to 13 items are collected, as shown in column (a) of Fig. 8. When the robot circles around, frequent occlusions happens in all the scenes. In the configurations, some items with the same class are closely placed together to make the data association even more challenging. We compared our method with three other commonly adopted techniques.

The first is Intersection Over Union (IOU), which calculates 2D box IOU for each candidate cuboid with existing cuboids. When trying to associate cuboids in a new frame, the IOU method back projects all existing cuboids in the local map back to this frame, and then each candidate cuboid is associated with the one that has the maximum IOU value. Another technique is Shared Map Points counting (SMP), which counts the shared map points between each candidate cuboid with all existing objects in the map, then the candidate cuboid is

Fig. 7: Mapping result (top view) for the KITTI odometry sequence 07. In addition to the black map points and green KeyFrames, detected cars are rendered as colored rectangles. 7 colores are repeatedly used to tint the cuboid for better visualization. A KeyFrame is selected to demonstrate the data association between cuboid observations and existing objects.

associated with the one that shares most map points with it. The next technique is Maximum Likelihood Estimation(MLE) [29], which simply associates each new cuboid to the closest one. In all 3 configurations, our method works reliably as shown in Table II. To perform fair comparisons, the final number of objects is filtered for each method. For example, if an object was observed for only a few frames, the object is ignored. Our method demonstrates superiority both before and after filtering.

4) Threshold setting: The association accuracy is shown in Fig. 6. For configuration 1 in Fig. 8, Υ is set to different values to demonstrate the impact of this threshold in data association. When Υ is 10, almost no new object is created even if the observation does not fit into any existing clusters. Consequently, the association can be wrong. The tracking is quickly failed due to the detrimental effect of false associations. On the other hand, when Υ is 3, too many new objects are created even if the observation belongs to an existing cluster. Consequently, some objects can be associated with more than 1 object in the pose graph. Although this does not break the tracking, the constraints provided by the object landmarks are weakened. Therefore, the strategy to set Υ is as follows. First, try some large value to not break the tracking and then gradually decrease the value to the best one.

C. KITTI Odometry Dataset

Lastly, to validate our data association method on a larger outdoor environment, we run our system on KITTI [31] odometry dataset. Most sequences in KITTI odometry benchmarks are static with a few dynamic objects such as cyclists. Our data association method performs well in all the sequences except 01, 02, and 04 which are highly dynamic. Data association in algorithm 1 is carried out whenever a new KeyFrame is created. The constant Υ is set to 6. The bundle adjustment of camera poses, cuboids, and points is performed locally in the LocalMapping thread and globally in the LoopClosing thread. Besides, the iteration cycles for data association and joint optimization are limited to 1 for efficiency.

1) Mapping With Cuboids: There are hundreds of cars and a few cyclists used as landmarks in the KITTI dataset. The computation cost of line 8 in Algorithm 1 which tests new cuboid fitting into existing clusters can be intimidating. To reduce the computation cost, only objects located in front of the camera are considered. This greatly improves the computation efficiency of the algorithm when the number of landmarks is large. To detect cuboids with true scale, the prior knowledge of camera height is used. The mapping result including the detected cuboids for sequence 07 is shown in Fig. 7. Other sequences are shown in Fig. 10.

2) Scale Drift and Pose Estimation: Because the cuboid detections are performed in each single frame, the scale of the cuboid measurement is consistent. When the cuboid measurements are incorporated in the bundle adjustment, they implicitly provide constraints for the scale. As a result, the scale drift of monocular SLAM is significantly reduced. It can be seen from Fig. 9 that, even without loop closure, the trajectory of the proposed method is much closer to the ground truth trajectory than the original ORB-SLAM2 (Mono). Duncan *et al.* [30] and CubeSLAM [15] and Dynamic-SLAM [12] also studied using objects to reduce monocular scale drift. We compare the RMSE of the Absolute Trajectory Error (ATE) [32] with all of them in Tab. III. Our method achieves the best localization accuracy in most sequences.

D. Loop Closing

Loop closing is essential in SLAM systems to reduce accumulated drift when a robot returns to a previously visited place. In this section, we show our cuboid landmarks can be used to assist loop closing. In the indoor Turtlebot3 experiment introduced in Section IV-B, the drift is insignificant. Because the correct associations of landmarks both observed by early KeyFrames and recent KeyFrames automatically formed a loop in the pose graph, the successful mapping result (Fig. 2) can be obtained even without loop closing. On the contrary, in some large KITTI datasets, the drift could be so prominent that the objects observed by recent KeyFrames formed different clusters from previous ones.

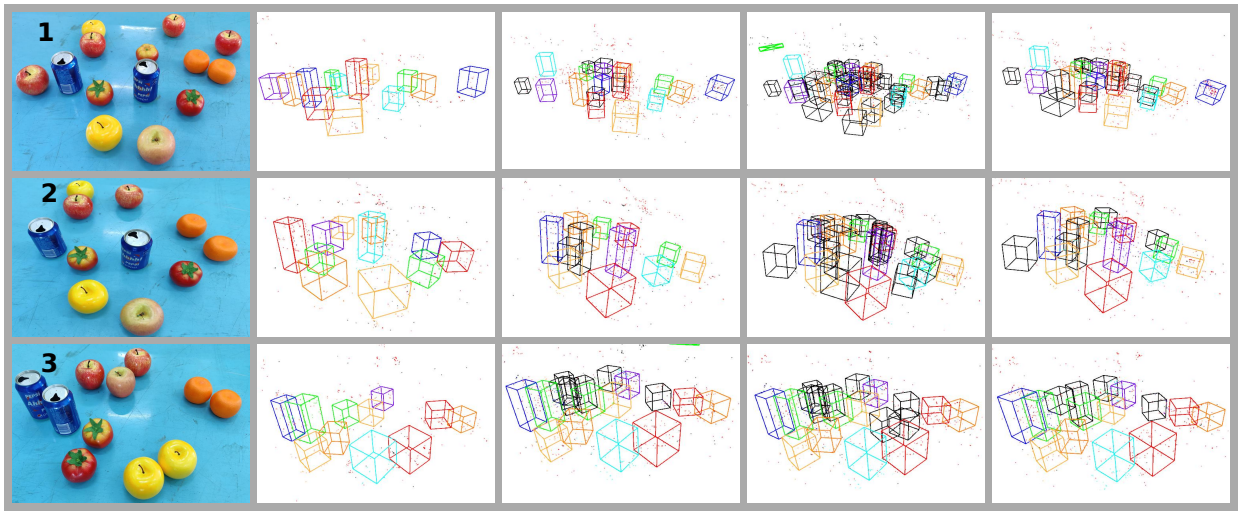
Traditional visual SLAMs detect loops based on appearances. OBR-SLAM2 detect loops by computing the difference of visual Bag of Words (Bow) of each KeyFrame. After a loop is detected, the similarity transformation matrix $S_{cm} \in sim(3)$ between current KeyFrame T_{cw} and the matched KeyFrame

TABLE II: Comparison of different method's data association results for 3 different configuration of items

Dataset	IOU		SMP		MLE		Ours	Ground Truth
	Before Filter	After Filter	Before Filter	After Filter	Before Filter	After Filter		
Conf. 1	25	19	43	29	33	28	13	14
Conf. 2	16	13	33	23	18	16	11	11
Conf. 3	20	17	26	19	24	15	11	11

TABLE III: Monocular Camera Pose Estimation Error on KITTI Odometry Benchmark

Sequence		00	03	05	06	07	08	09	10	Mean
ATE RMSE (m)	ORB-SLAM2(Mono) No LC	108.2	66.7	136.4	81.6	110.2	132.9	58.8	42.1	92.1
	Duncan *et al.* [30]	73.4	10.7	50.8	73.1	47.1	72.2	31.2	53.5	51.5
	CubeSLAM [15]	13.9	3.79	4.75	6.98	2.67	10.7	10.7	8.37	7.73
	Dynamic-SLAM [12]	4.436	0.828	5.724	12.455	1.823	27.487	9.285	8.909	8.86
	Ours	4.33	3.29	5.47	5.68	2.48	11.8	7.2	10.9	6.39



(a) 3 different configurations (b) Ours (c) IoU (d) SMP (e) MLE

Fig. 8: Cuboid mapping results for different methods, the first column (a) shows the picture of the configuration of items. The other columns from left to right, shows the result of Ours in column (b), Intersection over Union (IOU) in column (c), Shared map points counting (SMP) in column (d), Maximum likelihood estimation (MLE) in column (e) for different 3 configurations of items.

TABLE IV: Relative Pose Error of camera trajectory with or without cuboids during loop closing

KITTI sequences		00	05	07	08
RPE RMSE (m)	No Cuboids	0.239	0.269	0.104	0.149
	Cuboids	0.144	0.171	0.087	0.147

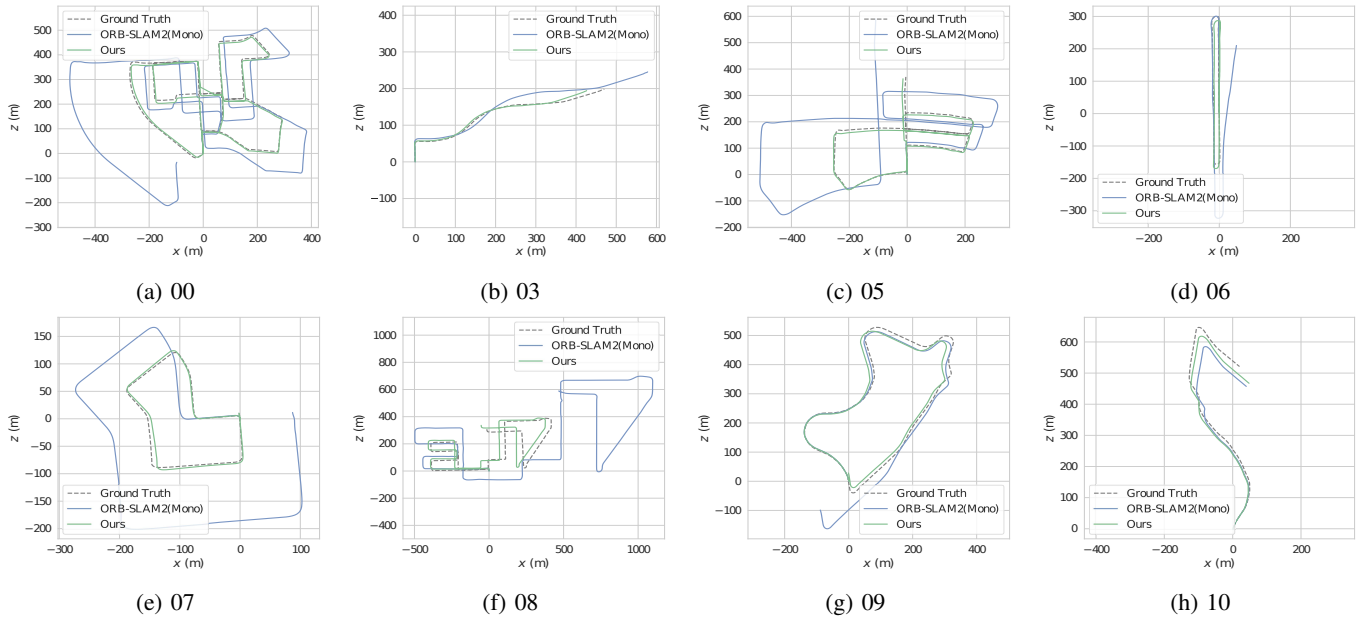
TABLE V: Runtime Breakdown For Our System

Component	Tracking	Cuboid Detection	Data Association	Joint Optimization
Time (ms)	33.5	87.9	26.2	284.7

T_{mw} is computed. Then the $sim(3)$ transformation S_{cw} can be obtained through $S_{cm}S_{mw}$, where S_{mw} is converted from T_{mw} with scale 1. We further project the objects observed in the current KeyFrame and the matched KeyFrame through S_{cw} and S_{mw} respectively. Finally, the data association is carried out under the same scale. It's the same motivation as fusing map points at both sides of the loops, more matches provide more constraints. Even better, the cuboids provide extra constraints for the scale. In the KITTI benchmark, loops only exist in sequences 00, 05, 07, and 08. To quantitatively analyze the effect of using cuboids during loop closing, we present the Relative Pose Error (RPE) of camera trajectory with cuboids in the first row of Tab. IV and RMSE without cuboids in the second row.

E. Time Analysis

Finally, we analyze the computation cost of our system. The system runs on Intel CPU i7-9700k at 3.6GHz except for the cuboid detection component, which runs on a separate Nvidia GPU RTX3090. The deep learning based cuboid detection component runs at about 11 frames per second. Although it does not add extra computation cost to the CPU, the frame rate of the tracking is slightly lowered to match the cuboid detection speed. Therefore, we manually set the frame rate to 11 in KITTI experiments. The data association and joint optimization components run efficiently in separate local mapping and loop closing threads, and they don't need to run in real-time. The inclusion of objects in joint optimization slightly increases the time of bundle adjustment. All the components average running times are presented in Table V.



(a) 00 (b) 03 (c) 05 (d) 06

(e) 07 (f) 08 (g) 09 (h) 10

Fig. 9: Camera trajectory comparison between ORB-SLAM2 (Mono) with our method on KITTI odometry benchmark. Black dotted lines are the ground truth. Blues lines are ORB-SLAM2 with significant scale drift. Green lines are Ours with scale drift reduced by incorporating cuboid measurements. Loop closing is disabled for both methods.

V. Conclusions

In this work, we have proposed a new monocular semantic object association algorithm using cuboid detections. We showed that cuboid detection can greatly benefit data association problems, which plays a vital role in semantic SLAM. The idea is based on the Dirichlet Process Mixture Model under two assumptions. First, most objects in the environment are static at least during a few local frames, and this allows the time index to be dropped when performing multi-frame data associations. Second, the measurement noise covariance is small. The cuboid detection accuracy is mainly affected by the 2D bounding box and image distortion. In the experiments, the 2D bounding box is accurate enough and the distortion is negligible. Thus this assumption is easily satisfied. The result is that the association problem reduces to a clustering problem. The main contribution of our work is combining cuboid detection with data association. Cuboid landmarks have properties like semantic class, location, and size, and thus are more distinguishable than pure geometric points. Experimental results showed that our method is more robust to ambiguous detections and false class predictions. The system runs about 11 frames per second on a Intel i7 CPU with seperate Nvidia GPU. Compared with other state of the art monocular SLAM methods, we observed over 17% overall improvement in terms of root mean squared error over translational components of absolute trajectory error on the public KITTI odometry dataset. In the future, we plan to study object reconstruction using a monocular camera and integrate the inference of the object shapes with SLAM optimization.

References

[1] A. Nüchter and J. Hertzberg, "Towards semantic maps for mobile robots," *Robot. Auton. Syst.*, vol. 56, no. 11, pp. 915–926, 2008.

[2] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 826–840, 2013.

[3] J. Ku, A. D. Pon, and S. L. Waslander, "Monocular 3d object detection leveraging accurate proposals and shape reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 11 867–11 876, 2019.

[4] G. Brazil and X. Liu, "M3d-rpn: Monocular 3d region proposal network for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Pattern Recognit.*, pp. 9287–9296, 2019.

[5] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. Kelly, and A. J. Davison, "Slam++: Simultaneous localisation and mapping at the level of objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 1352–1359, 2013.

[6] J. McCormac, R. Clark, M. Bloesch, A. Davison, and S. Leutenegger, "Fusion++: Volumetric object-level slam," in *Intl. Conf. on 3D Vision (3DV)*, pp. 32–41. IEEE, 2018.

[7] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 7263–7271, 2017.

[8] H. J. Levesque, "Knowledge representation and reasoning," *Annu. Rev. Comput. Sci.*, vol. 1, no. 1, pp. 255–287, 1986.

[9] X. Chen, A. Milioto, E. Palazzolo, P. Giguere, J. Behley, and C. Stachniss, "Suma++: Efficient lidar-based semantic slam," in *IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, pp. 4530–4537, 2019.

[10] S. Pillai and J. Leonard, "Monocular slam supported object recognition," *arXiv preprint arXiv:1506.01732*, 2015.

[11] F. Zhong, S. Wang, Z. Zhang, and Y. Wang, "Detect-slam: Making object detection and slam mutually beneficial," in *IEEE Winter Conf. Appl. of Comput. Vis. (WACV)*, pp. 1001–1010, 2018.

[12] L. Xiao, J. Wang, X. Qiu, Z. Rong, and X. Zou, "Dynamic-slam: Semantic monocular visual localization and mapping based on deep learning in dynamic environment," *Robot. Auton. Syst.*, vol. 117, pp. 1–16, 2019.

[13] A. Sharma, W. Dong, and M. Kaess, "Compositional scalable object slam," *IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021.

[14] L. Nicholson, M. Milford, and N. Sünderhauf, "Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam," *IEEE Trans. Robot. Autom.*, vol. 4, no. 1, pp. 1–8, 2018.

[15] S. Yang and S. Scherer, "Cubeslam: Monocular 3-d object slam," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 925–938, 2019.

[16] M. Z. Ikram and M. Ali, "A new data association method for 3-d object tracking in automotive applications," in *IEEE Radar Conf.*, pp. 1187–1191, 2014.

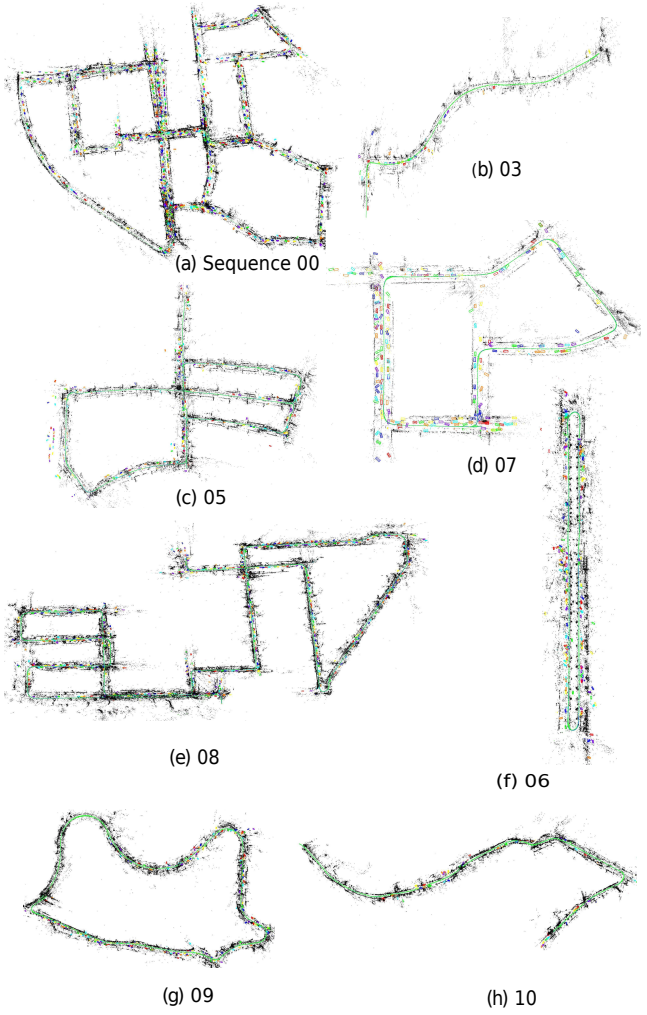


Fig. 10: Cuboid mapping results for KITTI odometry sequences. All the maps are resized to save space. The colored cuboids are so small compared to the full trajectory that they look as if they are dots.

[17] M. P. Muresan and S. Nedevschi, "Multi-object tracking of 3d cuboids using aggregated features," in *Proc. IEEE 15th Int. Conf. Intell. Comput. Commun. Process. (ICCP)*, pp. 11–18, 2019.

[18] S. A. Memon, "Smoothing data association for linear multi-target tracking in clutter," in *36th Int. Tech. Conf. Circuits Sys. Comput. Commun. (ITC-CSCC)*, pp. 1–4, 2021.

[19] B. Bićanić, I. Marković, and I. Petrović, "Multi-target tracking on riemannian manifolds via probabilistic data association," *IEEE Signal Process. Lett.*, vol. 28, pp. 1555–1559, 2021.

[20] S. L. Bowman, N. Atanasov, K. Daniilidis, and G. J. Pappas, "Probabilistic data association for semantic slam," in *IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 1722–1729, 2017.

[21] J. Elfring, S. van den Dries, M. Van De Molengraft, and M. Steinbuch, "Semantic world modeling using probabilistic multiple hypothesis anchoring," *Robot. Auton. Syst.*, vol. 61, no. 2, pp. 95–105, 2013.

[22] L. L. Wong, L. P. Kaelbling, and T. Lozano-Pérez, "Data association for semantic world modeling from partial views," *Int. J. Robot. Res.*, vol. 34, no. 7, pp. 1064–1082, 2015.

[23] J. Zhang, M. Gui, Q. Wang, R. Liu, J. Xu, and S. Chen, "Hierarchical topic model based object association for semantic slam," *IEEE Tran. Vis. Comput. Graphics*, vol. 25, no. 11, pp. 3052–3062, 2019.

[24] K. Jiang, B. Kulis, and M. Jordan, "Small-variance asymptotics for exponential family dirichlet process mixture models," *Adv. Neural. Inf. Process. Syst.*, vol. 25, pp. 3158–3166, 2012.

[25] J. M. Bernardo and A. F. Smith, *Bayesian theory*, vol. 405. John Wiley & Sons, 2009.

[26] G. Grisetti, R. Kümmerle, H. Strasdat, and K. Konolige, "g2o: A general framework for (hyper) graph optimization," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 9–13, 2011.

[27] B. Mu, S.-Y. Liu, L. Paull, J. Leonard, and J. P. How, "Slam with objects using a nonparametric pose graph," in *IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, pp. 4602–4609. IEEE, 2016.

[28] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, 2017.

[29] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics.*, ser. Intelligent robotics and autonomous agents. MIT Press, 2005.

[30] D. Frost, V. Prisacariu, and D. Murray, "Recovering stable scale in monocular slam using object-supplemented bundle adjustment," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 736–747, 2018.

[31] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 3354–3361, 2012.

[32] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 573–580, 2012.

Songlin Wei received the bachelor degree in software engineering from Xiamen University, Xiamen, China, in 2007. He is now pursuing the master degree with the School of Mechanical and Electrical Engineering, Soochow University. His research field includes Visual SLAM and AI Robotics.

Guodong Chen was born in 1983. He received the Ph.D. degree from the Harbin Institute of Technology, Harbin, China, in 2011. He is currently an Associate Professor with Soochow University. His research interests include robot vision and intelligent industrial robots.

Wenzheng Chi received her B.E. degree in automation from Shandong University, Jinan, China, in 2013 and the Ph.D. degree in Biomedical Engineering from the Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong, in 2017. She is currently an Associate Professor with the Robotics and Microsystems Center, School of Mechanical and Electric Engineering, Soochow University, Suzhou, China.

Zhenhua Wang received the Ph.D. degree from the Harbin Institute of Technology,Harbin, China, in 2009. He is currently an Professor with Soochow University. His research interests include intelligent industrial robots and its applications.

Lining Sun was born in Hegang, Heilongjiang Province, China in January 1964. In 1993, Prof. Sun received his Ph.D. degree in Engineering from the Mechanical Engineering Department of Harbin Institute of Technology (HIT), and joined HIT after graduation. He is a National Outstanding Youth Fund Winner, Changjiang Scholar Distinguished Professor by the Ministry of Education.